

Absolvování individuální odborné praxe

Individual Professional Practice in the Company

Prohlašuji, že jsem tuto bakalářskou práci vypracoval samostatně. Uvedl jsem všechny literární prameny a publikace, ze kterých jsem čerpal.

V Ostravě 6. května 2011

.....

Abstrakt

Tato bakalářská práce pojednává o průběhu mé odborné praxe, kterou jsem absolvoval ve firmě Cathedral Software, s.r.o. Uvádím zde profil firmy, ve které jsem praxi vykonával, spolu s úkoly, které jsem řešil. Následně popisuji postup řešení těchto úkolů. Ke konci jsou uvedeny znalosti a dovednosti, které jsem získal v průběhu studia a použil je při řešení, spolu s nově nabytými znalostmi z průběhu praxe. Na závěr hodnotím celou praxi, její průběh a přínos.

Klíčová slova: Odborná praxe, Cathedral Software, IntraWeb, Informační systém, MS SQL, .NET, Silverlight, 3D vizualizace

Abstract

This bachelor thesis is focused on my practical experience, which I passed in Cathedral Software, s.r.o. Here I present a profile of the company in which I worked, along with projects I've been working on. Consequently, I describe the process of solving these projects. Furthermore I sum up skills and knowledge I gained in the course of my studies, which I used at work, along with newly acquired skills gained from my practical experience. In the end I sum up the entire experience, its course and benefits.

Keywords: Professional practice, Practical experience, Cathedral Software, IntraWeb, Information system, MS SQL, .NET, Silverlight, 3D visualization

Seznam použitých zkratek a symbolů

IS	– Informační systém
MS	– Microsoft
.NET	– .NET Framework
SQL	– Structured Query Language
MS SQL	– Microsoft Structured Query Language Server
MySQL	– My Structured Query Language Server
XML	– Extensible Markup Language
XAML	– Extensible Application Markup Language
TCP	– Transmission Control Protocol
IP	– Internet Protocol
VCL	– Visual Components Library
GUI	– Graphical User Interface
RIA	– Rich Internet Application
ArisCAT	– Informační systém společnosti Cathedral Software s.r.o.

Obsah

1	Úvod	3
2	Zasazení praxe	4
2.1	Popis odborného zaměření firmy Cathedral Software, s.r.o.	4
2.2	Popis pracovního zařazení	4
3	Zadané úkoly v průběhu praxe	5
3.1	IS pro evidenci a sledování řešení požadavků údržby a rozvoje	5
3.2	Studie: Možnosti vývoje internetových aplikací	5
3.3	3D vizualizace výrobků	5
4	Zvolený způsob řešení zadaných úkolů	6
4.1	IS pro evidenci a sledování řešení požadavků údržby a rozvoje	6
4.1.1	Zadání	6
4.1.2	Použité technologie	6
4.1.3	Postup řešení	6
4.1.4	Využití projektu	7
4.2	Studie: Možnosti vývoje internetových aplikací	8
4.2.1	Zadání	8
4.2.2	Řešení	8
4.2.3	Využití projektu	8
4.3	3D vizualizace výrobků	9
4.3.1	Zadání	9
4.3.2	Použité technologie	9
4.3.3	Postup řešení	10
4.3.4	Využití projektu	12
5	Znalosti a dovednosti, které jsem využil v průběhu praxe	13
6	Znalosti a dovednosti, které jsem získal v průběhu praxe	14
7	Závěr	15
8	Reference	16

Seznam obrázků

1	Souřadnicové systémy 3D prostoru	11
---	--	----

1 Úvod

Cílem bakalářské práce je popsat průběh individuální odborné praxe ve firmě Cathedral Software, s.r.o. V následujících dvou kapitolách vás seznámím s firmou, ve které praxe probíhala a s úkoly, které jsem zpracovával. Dále uvádím stručný postup řešení těchto úkolů. Ke konci shrnuji použité a nově nabyté schopnosti a získané znalosti z průběhu praxe. Na závěr hodnotím průběh a přínos praxe.

2 Zasazení praxe

2.1 Popis odborného zaměření firmy Cathedral Software, s.r.o.

Společnost Cathedral Software, s.r.o.[1] se zaměřuje hlavně na vývoj informačních systémů. V této oblasti se jedná především o vývoj a dodávku úspěšného modulárního informačního systému ArisCAT[2], kterým se společnost zabývá již od roku 2001 a který využívají různé obchodní a výrobní firmy. ArisCAT je specifický především tím, že je optimalizován na zpracování obtížně konfigurovatelných výrobků, mezi něž patří například žaluzie, rolety, vrata a plastová okna. Dále se firma zabývá vývojem a následnou distribucí integrovaného modulu pro informační systém Helios Orange. Probíhá také implementace informačního systému pro výchovné ústavy. Dalším polem působnosti společnosti je tvorba zakázkového software, prodej značkové výpočetní techniky, správa a údržba počítačových sítí.

2.2 Popis pracovního zařazení

Po nástupu do firmy jsem byl seznámen s hlavním produktem firmy systémem ArisCAT a byla mi přidělena pozice samostatného vývojáře. Dle aktuálních potřeb firmy mi byly zadávány úkoly a v pravidelných intervalech byl kontrolován a konzultován postup prací. V případě obtíží jsem měl možnost konzultovat své problémy s dalšími zaměstnanci firmy.

3 Zadané úkoly v průběhu praxe

V průběhu praxe mi byly zadány celkem tři úkoly. Nejprve jsem se musel seznámit se systémem ArisCAT a následně jsem dostal první úkol, vytvořit webové rozhraní pro informační systém, který bude evidovat a zobrazovat průběh prací na jednotlivých požadavcích klientů. Po dokončení funkčnosti projektu a prvotním návrhu uživatelského rozhraní jsem dostal další úkol. Tím bylo zpracovat stručnou studii o možnostech vývoje webových aplikací. Po dokončení tohoto úkolu jsem dostal svůj poslední úkol, a tím bylo vytvořit program pro 3D vizualizaci výrobků ze systému ArisCAT.

3.1 IS pro evidenci a sledování řešení požadavků údržby a rozvoje

Prvním úkolem bylo vytvořit webové rozhraní pro informační systém, který by umožňoval tvorbu nových požadavků klientem a následně sledování průběhu jejich řešení. Protože většina aplikací ve firmě byla vytvořena v programovacím jazyce Delphi, měl jsem ho taktéž použít. Konkrétně jsem měl použít webový framework IntraWeb. Potřebná data jsem měl získávat a ukládat do firemní databáze MS SQL, ve které již byly vytvořeny potřebné tabulky, naplněné fungující aplikací, která zaznamenává čas, strávený pracovníky firmy na jednotlivých požadavcích klientů.

3.2 Studie: Možnosti vývoje internetových aplikací

Jako druhý úkol jsem zpracovával stručnou studii, pojednávající o možnostech vývoje webových aplikací. Studie měla informovat o jednotlivých programovacích jazycích, vývojových prostředích, cenách na pořízení těchto prostředí, možnostech automatického testování a práce s databázemi.

3.3 3D vizualizace výrobků

Třetím a posledním úkolem bylo vytvořit webovou aplikaci pro dynamické zobrazení 3D modelu výrobku ze systému ArisCAT. Data a textury výrobku, které jsem následně zpracoval a zobrazil, mi byly poslány formou XML souboru a bitmap přes TCP/IP rozhraní.

4 Zvolený způsob řešení zadaných úkolů

4.1 IS pro evidenci a sledování řešení požadavků údržby a rozvoje

4.1.1 Zadání

Zadání prvního úkolu bylo vytvořit webové rozhraní ve frameworku IntraWeb, které bude komunikovat s firemní databází na MS SQL Serveru. Webové rozhraní bude obsahovat přihlášení, registraci do systému, zaslání zapomenutého hesla na email, přehled stavu zadaných zpracovávaných úloh, zadávání nových úloh a přehled nově zadaných nepřevzatých úloh.

4.1.2 Použité technologie

Pro vytvoření webových stránek informačního systému a práci s daty jsem měl použít dvě technologie. Pro tvorbu webových stránek to byl framework IntraWeb[3], také označován jako VCL for the Web, to v překladu znamená knihovna vizuálních komponent pro Web. Výhodou tohoto frameworku je, že umožňuje tvorbu webových stránek za pomoci vizuálních komponent, které se umístí stejně jako při tvorbě desktopových aplikací na formulář. Při kompilaci se následně převede celý obsah stránky na html a javascriptový kód. Tento postup sice urychluje tvorbu webových stránek, ale programátor nemůže zajistit úroveň výsledného kódu.

Tuto technologii jsem měl požit ze dvou důvodů. Za prvé je již integrována v Code-Gear Rad Studiu, ve kterém jsem pracoval. Za druhé umožňuje tvorbu celého webového rozhraní pouze v kódu Delphi. Právě tento programovací jazyk je hlavním nástrojem firmy, a proto jsem ho měl použít i já, aby byl výsledek mé práce jednoduše čitelný a v budoucnu snadno modifikovatelný dalšími pracovníky.

Pro práci s daty jsem měl použít druhou technologii, kterou byl Microsoft SQL Server 2008. Na tomto serveru byla firemní databáze s potřebnými tabulkami a daty pro mou práci.

4.1.3 Postup řešení

Tímto úkolem započala moje praxe ve firmě, takže prvním krokem bylo seznámení se systémem ArisCAT, který je hlavním produktem firmy, a proto na něj navazuje a souvisí s ním velké množství dalších programů. Byl mi také předveden a vysvětlen program pro sledování průběhu práce zaměstnanců. V něm si zaměstnanec zvolil, na které úloze právě pracuje a následně byl měřen čas, strávený nad řešením úlohy. Data o úloze, její stav, postup práce a další informace byly uloženy v MS SQL databázi.

Tímto skončilo moje zasvěcení do problematiky a započala má práce. Zpočátku jsem si musel oživit práci v Delphi a naučit se pracovat s IntraWebem. Práce v IntraWebu je velmi podobná tvorbě desktopových aplikací, ale má i svoje specifika. Bohužel dokumentace k němu je značně strohá, proto jsem hlavně vycházel z příložených příkladů u Rad Studia. Když jsem byl připraven, začal jsem tím, že jsem prostudoval strukturu databáze, se

kterou jsem měl pracovat. Prošel jsem potřebné tabulky a zjistil vazby mezi nimi a jejich atributy.

Jako první jsem vytvořil přihlašování do systému, které obsahovalo navázání prvního spojení s databází. Adresu, šifrované heslo a další nastavení připojení na databázi jsem měl uložené v konfiguračním souboru, který umožňoval pohodlnou a rychlou změnu těchto parametrů v případě potřeby. Původně jsem řešil komunikaci s databází skrze SQL dotazy. To bylo později změněno, aby se zvýšila bezpečnost přístupu k databázi a bylo možné přistupovat k tabulkám pouze přes procedury a funkce. Vytvořil jsem si proto pro všechny potřebné operace s databází odpovídající procedury a funkce na SQL serveru, které jsem poté volal z programu.

Po dokončení přihlašování jsem pokračoval tvorbou stránky s přehledem zpracovávaných úloh. Stránka umožňovala vyhledávání a třídění podle rozličných atributů a pro zpřehlednění jsem ještě vytvořil základní a rozšířený pohled na vyhledaná data, kde podle zvoleného režimu byly zobrazeny pouze základní nebo všechny atributy. Pokud některý vyhledávací atribut mohl nabývat pouze určitých hodnot, byly relevantní hodnoty načteny z databáze podle daných uživatelů. Např. každý uživatel mohl vyhledávat jen podle pracovníků, kteří již pracovali na jeho požadavcích. Další stránkou bylo přidávání nových úloh. To umožňovalo vložení nové úlohy s jejím popisem, tato úloha byla následně zařazena do nově přijatých úloh, které mohl uživatel vyhledat na další stránce. Pokud byla úloha přijata, byla přesunuta do zpracovávaných úloh.

Tímto byla hotova většina webového rozhraní, zbývala už jen registrace a zaslání zapomenutého hesla. Registrace byla řešena tak, že po zadání požadovaných údajů byl poslán email s těmito údaji na adresu firmy, kde po kontrole těchto údajů byl daný zákazník registrován. Zapomenuté heslo funguje obdobně, po zadání potřebných údajů je zákazníkovi zasláno heslo na jeho email.

Po dokončení funkčnosti jsem se zaměřil na doladění mé práce a tvorbu GUI. Komponenty stránky jsem se snažil uspořádat tak, aby byly uživatelsky přívětivé a celkový styl stránek jsem ladil tak, aby zapadal do již vytvořené webové prezentace firmy. Po vytvoření prvotní verze byl následně celý projekt předán na zpracování grafikovi.

4.1.4 Využití projektu

Sledování průběhu řešení požadavků klientů není v současné době prioritou firmy, proto bylo nasazení projektu pozastaveno. Věřím ale, že až bude tato problematika znovu aktuální, tak po dokončení grafické stránky projektu, bude možné toto řešení využít.

4.2 Studie: Možnosti vývoje internetových aplikací

4.2.1 Zadání

Zpracujte stručnou studii na téma možnosti vývoje webových aplikací. Studie bude obsahovat přehled programovacích jazyků, jejich vývojových prostředí, ceny těchto prostředí, možnosti automatického testování a možnosti spolupráce s databázovými stroji, zejména MS SQL a MySQL.

4.2.2 Řešení

Při zpracovávání tohoto materiálu jsem se zaměřil na šest nejpoužívanějších programovacích jazyků a technologií, které jsem rozdělil do dvou základních skupin. Technologie pro vývoj webových aplikací, mezi které jsem zařadil jazyky PHP, Java, C#(ASP.NET), Python a technologie pro vývoj RIA aplikací, do kterých jsem zařadil Silverlight a Adobe Flash.

U každého jsem uvedl stručný popis, jeho vlastnosti, výhody a nevýhody, možnosti práce s databázemi a možnosti automatického testování. Nakonec jsem uvedl různé varianty vývojových prostředí. Snažil jsem se najít jak zástupce open source, tak placeného softwaru, u placených prostředí jsem pak uvedl stručný přehled různých druhů licencí a verzí.

4.2.3 Využití projektu

Tento materiál slouží jako shrnutí a stručný popis možností dalšího vývoje firmy v oblasti webových aplikací. Byl již využit při rozhodování, jaké technologie použít při řešení třetího úkolu a jistě bude využíván i v budoucnu, do doby kdy bude ještě aktuální.

4.3 3D vizualizace výrobků

4.3.1 Zadání

Vytvořte Silverlight aplikaci, která bude umožňovat zobrazení 3D modelu výrobku, s možností zobrazení kót a zakomponování tohoto modelu do fotografie. Aplikace bude získávat data o výrobku ze systému ArisCAT pomocí TCP/IP, data zpracuje a vytvoří dle nich 3D model reálného produktu.

4.3.2 Použité technologie

Pro tvorbu jsem dle zadání použil framework Silverlight[4], se kterým jsem již dříve pracoval. Silverlight umožňuje tvorbu interaktivních webových, desktopových a mobilních aplikací. Může pracovat na různých webových prohlížečích, zařízeních a operačních systémech. Právě pro jeho interoperabilitu byl Silverlight zvolen pro tento projekt. Programování Silverlight aplikací se skládá ze dvou částí, a to programování v XAMLu a v kódové části. Při tvorbě aplikace jsem především používal kódovou část, protože jsem potřeboval vše zpracovávat dynamicky.

Další technologií, kterou jsem použil, je Balder[5]. Jedná se o 3D grafický engine, zaměřený na Silverlight, Windows Phone 7, XNA a OpenGL. Silverlight neumožňuje práci s 3D grafikou, proto bylo nutné najít řešení mimo jeho hranice. Při studiu tohoto problému jsem narazil na open source projekt Balder, který umožňoval vše, co jsem potřeboval. Pomocí tohoto projektu jsem byl schopen nejenom zobrazit libovolný 3D objekt, ale i tento objekt od základu vytvořit a následně s ním provádět různé interakce. Jediným problémem bylo potažení objektu texturou, kde bylo potřeba rozdělit každý povrch na trojúhelníky. Bohužel to již Balder neumožňuje, a proto je nutné určit vždy 3 body pro jeden trojúhelník textury ručně.

To nás přivádí k třetí technologii, kterou jsem použil. Po studiu triangulace a technik, jak dosáhnout rozdělení každého dvourozměrného obrazce na trojúhelníky, jsem stále nemohl najít vhodné řešení a už jsem se chystal zkusit vytvořit vlastní funkce pro triangulaci. Nakonec jsem našel Poly2tri[6]. Jedná se o open source knihovnu, která je přeložená do různých programovacích jazyků (např. C++, C#, Java, Python, apod.) a umožňuje Delaunayho triangulaci, po zadání bodů útvaru provede jeho triangulaci na jednotlivé trojúhelníky. Posledním problémem bylo, že Poly2tri neobsahuje verzi pro Silverlight, naštěstí nevyužívá žádných knihoven, které v Silverlight nejsou, proto jen stačilo zkompileovat knihovnu do Silverlight Class Library.

Pomocí těchto tří technologií jsem byl již schopen programově vytvořit, potáhnout texturou a zobrazit v podstatě libovolný 3D objekt.

4.3.3 Postup řešení

Prvním krokem při řešení tohoto projektu bylo analyzovat, jak a zda je vůbec možné splnit všechny požadavky zadání. Při zkoumání tohoto problému se ukázalo, že Silverlight podporuje pouze perspektivní 3D, to znamená, že umožňuje naklápění a otáčení objektu po osách, ve snaze vytvořit slabou iluzi 3D prostoru. Samostatný objekt, který by se pohyboval v prostoru, již vytvořit nedokáže.

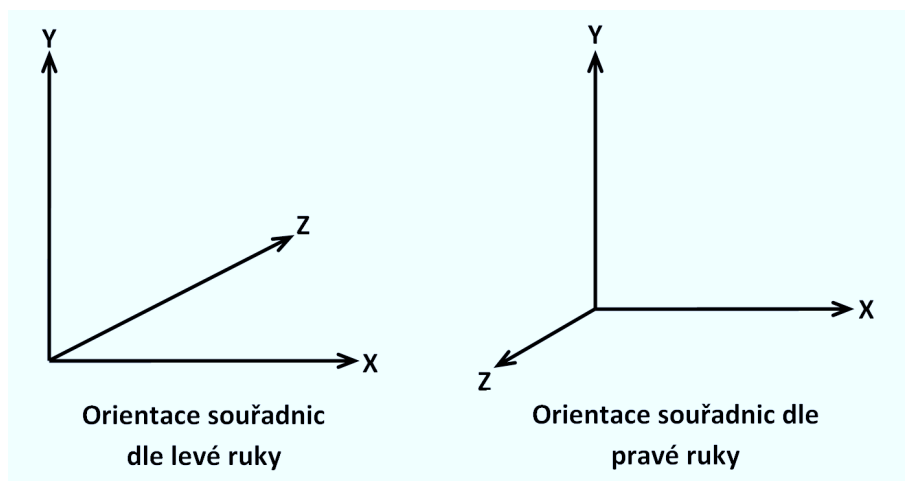
Bylo proto potřeba najít jiný nástroj pro zobrazení 3D grafiky nebo přehodnotit zadání projektu. Našel jsem několik nástrojů, které 3D v Silverlightu umožňovaly, ale byly placené nebo nebyly dostatečné pro naše potřeby. Nakonec jsem narazil na stánkách CodePlexu na projekt Balder, který splňoval všechny mé požadavky. Umožňoval relativně snadno a efektivně pracovat s objekty v 3D scéně. Obsahoval jak objekty, tak kamery i různé druhy světél. Předností, které jsem nejvíce využil, byla tvorba 3D objektů přímo z kódu. Balder měl ale i svá omezení, nepodporoval všechny formáty obrázků, textury musely mít rozměr strany ve velikosti mocniny dvou, byl stále ještě ve fázi vývoje, atd. Přednosti ale převyšovaly omezení a obsahoval již dostatek funkcí, proto jsem tento engine zvolil pro svůj projekt. Vystaly ale dva problémy, Balder neumožňoval psát text, proto bylo nutné vymyslet, jak kótovat výrobky a také potřeboval mít každou plochu rozdělenou na trojúhelníky, aby ji mohl pokrýt texturou.

Kótování jsem po konzultaci vyřešil vytvořením vlastních číslic a značek pro kóty, kde jsem využil toho, že Balder mohl kreslit i jednotlivé čáry. Kóty pak byly vytvořeny z čar a poskládaných digitálních číslic.

Vyřešení problému s triangulací bylo již horší. Všechna řešení, která jsem našel, byla buď licencovaná, nebo nevhodná pro použití a k vytvoření vlastního algoritmu pro triangulaci je potřeba obrovská znalost této problematiky a ladění algoritmu může trvat velmi dlouho. Po několikadenním studiu triangulace a hledání řešení se mi nakonec podařilo najít knihovnu Poly2Tri. Ta zvládla vše, co jsem potřeboval. Po zadání jednotlivých bodů útvaru provedla triangulaci a vrátila vždy tři body každého trojúhelníku, na který byl daný polygon rozdělen.

Poté, co byly všechny problémy, které mohly ohrozit dokončení projektu, vyřešeny, jsem mohl začít s jeho tvorbou. Projekt jsem si rozdělil na jednotlivé části, kterými musela data projít, abych byl nakonec schopen vytvořit jejich 3D reprezentaci.

První částí byla TCP/IP komunikace se systémem ArisCAT. Komunikace probíhala asynchronně přes sockety. Jako první byl poslán počet souborů, který se bude předávat, následně se vždy poslala velikost souboru a pak zmíněný soubor. Jako první soubor jsem obdržel XML soubor s daty, ten obsahoval údaje o kostře výrobku, profily, které se připojovaly na kostru, údaje o texturách a kóty. Kostra a profily byly definovány body a vektory mezi nimi. U kostry byla ještě informace, které profily a jak natočené patří na příslušné vektory. Dále soubor obsahoval informace o texturách a prostorech, které mají vyplňovat. Všechny hodnoty byly pouze relativní a bylo nutné dopočítat jejich reálné pozice dle zadaných kót. Pokud nebyl výrobek dostatečně okótován, byl uživatel informován, které body není program schopen dopočítat. Tento systém byl použit, aby se po vytvoření nákresu výrobku v systému ArisCAT daly zadat skutečné rozměry výrobku až při jeho koupi. Dalšími soubory již byly textury a fotka, na kterou měl být výrobek



Obrázek 1: Souřadnicové systémy 3D prostoru

případně umístěn. Následovalo zpracování obdržených dat, což zahrnovalo přepočítání počátečních bodů jednotlivých nákrešů, získání jejich reálných poloh, připojení profilů na kostru a přepočítání souřadnic do 3D prostoru. Programování tohoto zpracování bylo značně náročné, protože jsem se musel orientovat v přeměně 2D nákrešů v 3D objekt, kde navíc bylo nutné zjistit, jaké reálné rozměry daný objekt má. Převáděl jsem 2D nákresey s počátečním bodem v levém horním rohu na objekt v 3D prostoru, který byl orientovaný dle levé ruky viz. Obrázek 1.

Celý výrobek byl rozdělen na jednotlivé díly, a to profily na vektoru kostry a výplně s texturou, které byly nakonec vloženy do jediného uzlu, který je spojoval do celkového výrobku. Toto bylo nutné, protože jeden objekt mohl mít pouze jednu texturu.

Po získání reálných 3D dat jsem předával data jako pole částí, kde každá část měla svou texturu a plochy, které se skládaly z jednotlivých bodů.

Následně bylo potřeba tyto plochy ztriangulovat, aby se daly potáhnout texturou. Na to sloužila další část programu, ta postupně zpracovávala každou část výrobku, vzala jednotlivé povrchy a dle jejich natočení vybrala pro všechny body dvě souřadnice z původních tří, tím si je převedla na 2D polygony, které ztriangulovala a následně již vrátila posloupnost bodů, mezi kterými se jednotlivé trojúhelníky nachází a zbývající souřadnice se dohledala dle předchozích dvou.

Po provedení triangulace zbývalo už jen zobrazit konečný objekt. To obstarávala poslední část programu. Ta jako první vzala jednotlivé body, následně je spojila hranami a poté mezi nimi vymodelovala jednotlivé trojúhelníky, ze kterých poskládala všechny povrchy a těm nakonec přiřadila textury. Vždy se modelovala jedna část výrobku po druhé a následně se umístila do společného uzlu, který zapouzdřoval celý výrobek. Tento uzel se poté přiřadil na 3D plátno Balderu a byl zobrazen.

Program měl umožňovat zakomponovat výrobek do fotografie, proto bylo možné nahrát na pozadí zmíněnou fotografii a zvětšovat, zmenšovat a různě natáčet daný vý-

robek. Toho bylo možné docílit myší nebo použitím ovládacích tlačítek ve spodní části programu.

Další částí výrobku bylo zobrazení kót. Ty v případě, že měly být zobrazeny, byly přiřazeny jako další část výrobku do slučovacího uzlu.

Ve finální podobě programu si mohl uživatel prohlédnout, jak bude skutečný výrobek proporcionálně vypadat, mohl si nechat zobrazit kóty výrobku a následně si daný výrobek zakomponovat do fotografie a podívat se, jak bude ve skutečnosti přibližně vypadat ve zvoleném prostředí.

4.3.4 Využití projektu

Tento projekt navazuje na poslední vylepšení systému ArisCAT, kterým je možnost grafického návrhu výrobku. Po otestování bude implementován společně se systémem ArisCAT a umožní zobrazit reálnou podobu finálního produktu.

5 Znalosti a dovednosti, které jsem využil v průběhu praxe

V průběhu praxe jsem využil různé schopnosti a dovednosti, které jsem se naučil v průběhu studia. Při práci na prvním úkolu, kde jsem tvořil webové rozhraní pro informační systém, jsem těžil hlavně ze znalostí, nabytých při studiu databází. Byl jsem schopen bez problémů napsat všechny SQL dotazy a snadno si osvojit psaní procedur a funkcí v MS SQL. Dále jsem využil znalostí, jak správně navrhnout uživatelské rozhraní.

U posledního úkolu, kde jsem zobrazoval 3D model výrobku v Silverlight, jsem využil toho, že jsem měl možnost v Silverlighu již dříve pracovat. Bohužel předměty, které se zabývaly tvorbou 3D grafiky, jsem neabsolvoval, proto jsem se s touto problematikou musel seznámit sám.

U všech úkolů jsem využil i programátorské návyky, které jsem si při studiu osvojil, např. jak psát čistý a přehledný zdrojový kód.

6 Znalosti a dovednosti, které jsem získal v průběhu praxe

Za dobu absolvování praxe jsem se naučil mnoha praktickým a užitečným schopnostem a dovednostem. Naučil jsem se pracovat s IntraWebem, s databází MS SQL, prohloubil jsem své znalosti práce v Silverlight a naučil jsem se, jak v něm zobrazit 3D grafiku.

Kromě těchto odborných znalostí jsem se naučil efektivně hledat řešení problémů a spojení různých technologií tak, aby mi poskytly všechny potřebné nástroje pro implementaci všech požadavků zaměstnavatele.

Získal jsem také cenné zkušenosti, jak práce na projektech v praxi probíhá a jak se pohybovat v kolektivu dalších zaměstnanců.

7 Závěr

Jsem velmi rád, že jsem odbornou praxi absolvoval, protože jsem si zdokonalil nejenom své odborné znalosti a zkušenosti, které v budoucnu určitě využiji, ale hlavně jsem poznal reálné pracovní prostředí, které pouhým studiem poznat nelze. Uvědomil jsem si, že mi škola poskytla kvalitní základ, na kterém mohu dále stavět a rozvíjet své profesní znalosti tak, abych byl schopen splňovat dnešní požadavky neustále se měnícího pracovního trhu.

8 Reference

- [1] Internetové stránky Cathedral Software s.r.o. - [online]
URL: <<http://www.cathedral.cz>> [cit. 27.4.2011]
- [2] Internetové stránky IS ArisCAT - [online]
URL: <<http://www.cathedral.cz/cz/p/informacni-system-ariscat>> [cit. 27.4.2011]
- [3] Internetové stránky IntraWebu - [online]
URL: <<http://www.atozed.com/intraweb>> [cit. 27.4.2011]
- [4] Internetové stránky Silverlightu - [online]
URL: <<http://www.silverlight.net>> [cit. 27.4.2011]
- [5] Internetové stránky 3D enginu Balder - [online]
URL: <<http://balder.codeplex.com>> [cit. 27.4.2011]
- [6] Internetové stránky knihovny Poly2tri - [online]
URL: <<http://code.google.com/p/poly2tri>> [cit. 27.4.2011]